UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/762,814	01/22/2004	David Bau	ORACL-01388US1	6120

80548 7590 08/05/2008

Fliesler Meyer LLP
650 California Street
14th Floor
San Francisco, CA 94108

EXAMINER	
CHEN, QING	
ART UNIT	PAPER NUMBER
2191	
MAIL DATE	DELIVERY MODE
08/05/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/762,814	**Applicant(s)** BAU, DAVID	
	Examiner Qing Chen	**Art Unit** 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 May 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-65* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-65* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is in response to the RCE filed on May 6, 2008.

2. **Claims 1-65** are pending.

3. **Claims 1, 15, 18, 19, 23, 37, 40, 41, 44, 58, 61, 62, and 65** have been amended.

4. **Claim 66** has been cancelled.

5. The objection to the title is withdrawn in view of Applicant's amendments to the title.

6. Applicant has failed to address the objection to the specification due to the use of trademarks. Accordingly, this objection is maintained and further explained below.

7. The 35 U.S.C. § 101 rejections of Claims 23-39, 44-64, and 66 are withdrawn in view of Applicant's arguments and amendments to the claims or cancellation of the claim. However, the 35 U.S.C. § 101 rejections of Claims 1-22, 40-43, and 65 are maintained in view of Applicant's arguments and amendments to the claims and further explained below.

Response to Amendment

Specification

8. The title of the invention is objected to because the trademark or trade name JAVA should be accompanied with an appropriate designation symbol, *e.g.*, ™ or ®.

Appropriate correction is required.

9. The abstract of the disclosure is objected to because it should avoid using phrases which can be implied, such as, "The disclosure concerns," "The disclosure defined by this invention," "The disclosure describes," etc. Hence, the sentences "This description is not intended [...] Other

features, aspects, and objects of the invention […]" should be deleted. Correction is required. See MPEP § 608.01(b).

10. The disclosure is objected to because the attorney docket numbers should be deleted on page 1, paragraphs [0003], [0005], and [0006].

Appropriate correction is required.

11. The disclosure is objected to because it contains an embedded hyperlink and/or other form of browser-executable code on page 1, paragraph [0007]. Applicant is required to delete the embedded hyperlink and/or other form of browser-executable code. See MPEP § 608.01.

12. The use of trademarks, such as JAVA, has been noted in this application. Trademarks should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, *e.g.*, ™ or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, *e.g.*, "the JAVA programming language").

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner, which might adversely affect their validity as trademarks.

13. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the

following is required: “[M]achine readable medium” should read -- machine readable storage medium -- in Claims 44-64.

Claim Objections

14. **Claims 2-22, 24, 28, 37-41, 45, 49, and 58-60** are objected to because of the following informalities:

- **Claims 2-14, 16, 17, and 20-22** recite the category of invention “[t]he system.” Applicant is advised to change this limitation to read “[t]he computer-implemented system” for the purpose of providing it with proper explicit antecedent basis.
- **Claims 2, 24, and 45** contain a typographical error: “[A] Java web services method” should presumably read -- a Java web service method --.
- **Claims 6, 28, and 49** contain a typographical error: “[A] immovable value” should read -- an immovable value --.
- **Claim 12** recites the limitation “the data.” Applicant is advised to change this limitation to read “the XML data” for the purpose of providing it with proper explicit antecedent basis.
- **Claims 13 and 14** depend on Claim 12 and, therefore, suffer the same deficiency as Claim 12.
- **Claim 15** contains a typographical error: “[A]n Java object” should read -- a Java object --.
- **Claims 15, 37, and 58** contain a typographical error: “[T]he source and target type” should read -- the source and target types --.

- **Claims 15, 37, and 58** recite the limitation “the Java type.” Applicant is advised to change this limitation to read “the Java type system” for the purpose of providing it with proper explicit antecedent basis.
- **Claims 16 and 17** depend on Claim 15 and, therefore, suffer the same deficiency as Claim 15.
- **Claims 38 and 39** depend on Claim 37 and, therefore, suffer the same deficiency as Claim 37.
- **Claims 59 and 60** depend on Claim 58 and, therefore, suffer the same deficiency as Claim 58.
- **Claims 18, 19, 40, and 41** contain a typographical error: “[I]s capable of […]” should read -- and is capable of […] --.
- **Claim 20** contains a typographical error: “[A] hierarchical structure” should be written on the same line.
- **Claim 28** contains a typographical error: The repeated colon after the word “comprising” should be deleted.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

15. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

16. **Claims 1-65** are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claims 1, 15, 18, 19, 23, 37, 40, 41, 44, 58, 61, 62, and 65 recite the limitation of a common Java™ type that provides XML-oriented data manipulation. The subject matter is not properly described in the application as filed, since the specification only discloses an XML type that provides a number of XML-oriented data manipulations *(see page 5, paragraph [0032])*. The specification lacks disclosure on a common Java™ type that provides XML-oriented data manipulation. Because the specification does not adequately support the claimed subject matter, it would not reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claims 2-14 depend on Claim 1 and, therefore, suffer the same deficiency as Claim 1.

Claims 16 and 17 depend on Claim 15 and, therefore, suffer the same deficiency as Claim 15.

Claims 20-22 depend on Claim 19 and, therefore, suffer the same deficiency as Claim 19.

Claims 24-36 depend on Claim 23 and, therefore, suffer the same deficiency as Claim 23.

Claims 38 and 39 depend on Claim 37 and, therefore, suffer the same deficiency as Claim 37.

Claims 42 and 43 depend on Claim 41 and, therefore, suffer the same deficiency as Claim 41.

Claims 45-57 depend on Claim 44 and, therefore, suffer the same deficiency as Claim 44.

Claims 59 and 60 depend on Claim 58 and, therefore, suffer the same deficiency as Claim 58.

Claims 63 and 64 depend on Claim 62 and, therefore, suffer the same deficiency as Claim 62.

17. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

18. **Claims 1-65** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 1, 2, 4, 7-9, 15, 18, 19, 23, 24, 26, 29-31, 37, 40, 41, 44, 45, 47, 50-52, 58, 61, 62, and 65 contain the trademark or trade name JAVA. When a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of the 35 U.S.C. 112, second paragraph. *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, the use of a trademark or trade name in a claim to identify or describe a material or product (in the present case, a specific programming language) would not only render a claim indefinite, but would also constitute an improper use of the trademark or trade name.

Claims 3, 5, 6, and 10-14 depend on Claim 1 and, therefore, suffer the same deficiency as Claim 1.

Claims 16 and 17 depend on Claim 15 and, therefore, suffer the same deficiency as Claim 15.

Claims 20-22 depend on Claim 19 and, therefore, suffer the same deficiency as Claim 19.

Claims 25, 27, 28, and 32-36 depend on Claim 23 and, therefore, suffer the same deficiency as Claim 23.

Claims 38 and 39 depend on Claim 37 and, therefore, suffer the same deficiency as Claim 37.

Claims 42 and 43 depend on Claim 41 and, therefore, suffer the same deficiency as Claim 41.

Claims 46, 48, 49, and 53-57 depend on Claim 44 and, therefore, suffer the same deficiency as Claim 44.

Claims 59 and 60 depend on Claim 58 and, therefore, suffer the same deficiency as Claim 58.

Claims 63 and 64 depend on Claim 62 and, therefore, suffer the same deficiency as Claim 62.

Claims 1, 15, 18, 19, 23, 37, 40, 41, 44, 58, 61, 62, and 65 recite the limitation "an XML type […] implements a common Java type that provides XML-oriented data manipulation."

There is awkward claim language because it is unclear to the Examiner whether the XML type or the common Java™ type is providing the XML-oriented data manipulation. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “an XML type […] that provides XML-oriented data manipulation and implements a common Java type” for the purpose of further examination.

Claims 2-14 depend on Claim 1 and, therefore, suffer the same deficiency as Claim 1.

Claims 16 and 17 depend on Claim 15 and, therefore, suffer the same deficiency as Claim 15.

Claims 20-22 depend on Claim 19 and, therefore, suffer the same deficiency as Claim 19.

Claims 24-36 depend on Claim 23 and, therefore, suffer the same deficiency as Claim 23.

Claims 38 and 39 depend on Claim 37 and, therefore, suffer the same deficiency as Claim 37.

Claims 42 and 43 depend on Claim 41 and, therefore, suffer the same deficiency as Claim 41.

Claims 45-57 depend on Claim 44 and, therefore, suffer the same deficiency as Claim 44.

Claims 59 and 60 depend on Claim 58 and, therefore, suffer the same deficiency as Claim 58.

Claims 63 and 64 depend on Claim 62 and, therefore, suffer the same deficiency as Claim 62.

Claim 35 recites the limitation "the data types and ranges of the XML type." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “data types and ranges of the XML type” for the purpose of further examination.

Claim 36 recites the limitation “the constraints on the XML type.” There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “constraints on the XML type” for the purpose of further examination.

Claims 40, 41, 43, 61, 62, and 64 recite the limitation “the XML type.” There is insufficient antecedent basis for this limitation in the claims. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “an XML type” for the purpose of further examination.

Claims 40, 58, 61, and 62 recite the limitation “the XML schema.” There is insufficient antecedent basis for this limitation in the claims. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “an XML schema” for the purpose of further examination.

Claim Rejections - 35 USC § 101

19. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

20. **Claims 1-22, 40-43, and 65** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 1-22 are directed to computer-implemented systems. However, the recited components of the computer-implemented systems appear to lack the necessary physical components (hardware) to constitute a machine or manufacture under § 101. Although the claims recite the systems as being implemented on a computer, however, such recitation can be construed to be an intended use of the computer rather than a physical component of the systems. Therefore, these claim limitations can be reasonably interpreted as computer program modules—software *per se*. The claims are directed to functional descriptive material *per se*, and hence non-statutory.

The claims constitute computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical "things." They are neither computer components nor statutory processes, as they are not "acts" being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program's functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program's functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

Claims 40-43 are directed to a method. However, the recited steps of the method are held to be non-statutory subject matter because the recited steps of the method are (1) not tied to another statutory class (such as a particular apparatus) or (2) not transforming the underlying subject matter (such as an article or materials) to a different state or thing. Applicant is advised to amend the claims to recite "[a] computer-implemented method" in order to overcome the § 101 rejections.

Claim 65 contains "means-plus-function" limitations and appears to be a system. However, it is noted that the specification does not disclose any specific corresponding structure or equivalents thereof. The recited means appear to lack the necessary physical components (hardware) to constitute a machine or manufacture under § 101. Therefore, these claim limitations can be reasonably interpreted as computer program modules—software *per se*. The claims are directed to functional descriptive material *per se*, and hence non-statutory.

The claims constitute computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical "things." They are neither computer components nor statutory processes, as they are not "acts" being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program's functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program's functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

Claim Rejections - 35 USC § 102

21. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

22. **Claims 1-65** are rejected under 35 U.S.C. 102(e) as being anticipated by **US 7,155,705 (hereinafter "Hershberg")**.

As per **Claim 1**, Hershberg discloses:

- an XML data *(see Figure 1A: 118; Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118 ...")*;

- an XML schema which defines the XML data *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.")*;

- an XML type, which corresponds to the XML schema, that provides XML-oriented data manipulation and implements a common Java™ type, wherein the XML type allows the combination of XML and Java type systems and is capable of accessing and manipulating the XML data from within Java™ *(see Figure 1A: 116; Column 7: 31-38, "... the developer generates a modified JAVA source code file 112 based on the JAVA source code file 104 by inserting into a comment statement 114, for each class of data objects to be exchanged with other applications, data exchange configuration information in the form of XML configuration information 116 that indicates a relationship between data object classes and XML elements."; and 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5."; Column 8: 33-37, "Thus the XML configuration information 116 in comment 114 in the example of lines 3 through 4 indicates that the public class Employee is to be mapped to the root element "Employee" in the XML grammar defined in employee.dtd.")*; and

- a compiler capable of generating the XML type from the XML schema *(see Column 6: 37, "JAVA compilers ..."; Column 10: 12-16, "An example XML document with statements using the elements defined in the DTD file are given in Table 4 for two employees, John Smith and Jane Doe having employee identification numbers of 73645112 and 1, respectively.").*

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the compiler is capable of generating the XML type based on the definition of a Java™ web service method *(see Column 6: 37, "JAVA compilers ..."; Column 10: 12-16, "An example XML document with statements using the elements defined in the DTD file are given in Table 4 for two employees, John Smith and Jane Doe having employee identification numbers of 73645112 and 1, respectively.")*.

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the compiler is capable of generating the XML type based on a definition file *(see Column 6: 37, "JAVA compilers ..."; Column 10: 12-16, "An example XML document with statements using the elements defined in the DTD file are given in Table 4 for two employees, John Smith and Jane Doe having employee identification numbers of 73645112 and 1, respectively.")*.

As per **Claim 4**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the compiler is capable of compiling a Java™ project into one or more regular Java™ types *(see Column 6: 37, "JAVA compilers ..."; Column 10: 12-16, "An example XML document with statements using the elements defined in the DTD file are given in Table 4 for two employees, John Smith and Jane Doe having employee identification numbers of 73645112 and 1, respectively.")*.

As per **Claim 5**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type can be a movable cursor, capable of reading anywhere within the XML data *(see Column 7: 39-42, "The XML configuration information 116 is inserted into the comment statement associated with a definition statement for a class of data objects, including the attributes of the class.").*

As per **Claim 6**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type can be an immovable value, capable of referencing a fixed part of the XML data *(see Column 7: 39-42, "The XML configuration information 116 is inserted into the comment statement associated with a definition statement for a class of data objects, including the attributes of the class.").*

As per **Claim 7**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type can be shared among multiple Java™ components *(see Column 7: 39-42, "The XML configuration information 116 is inserted into the comment statement associated with a definition statement for a class of data objects, including the attributes of the class.").*

As per **Claim 8**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type is capable of updating the XML data within Java™ *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.").*

As per **Claim 9**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type is capable of accessing and updating Java™ data using Java™ type methods *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.").*

As per **Claim 10**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type is capable of accessing and updating a database *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.").*

As per **Claim 11**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- the XML type is capable of a number of XML data operations, which include: querying XML data, transforming between XML types, and iterating over XML data document *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.").*

As per **Claim 12**, the rejection of **Claim 1** is incorporated; and Hershberg further discloses:

- an XML schema capable of defining the legal types of the XML data, which include constraints on data types and ranges of the XML data; and constraints on the data types and ranges of the XML type *(see Column 9: 35-42).*

As per **Claim 13**, the rejection of **Claim 12** is incorporated; and Hershberg further discloses:

- the compiler is capable of generating constraints on the XML type from the XML schema on legal types of the XML data *(see Column 6: 37, "JAVA compilers ..."; Column 10: 12-16, "An example XML document with statements using the elements defined in the DTD file are given in Table 4 for two employees, John Smith and Jane Doe having employee identification numbers of 73645112 and 1, respectively.").*

As per **Claim 14**, the rejection of **Claim 12** is incorporated; and Hershberg further discloses:

- the constraints on the XML type are capable of validating the XML type *(see Column 9: 44-55, "The XML DTD statement in line 1 of Table 3 indicates that the root element, the first element in the DTD file, is called Employee and that the Employee element must include the sub-element LastName only once.").*

As per **Claim 15**, Hershberg discloses:

- a Java™ data *(see Column 7: 49 and 50, "For example, Table 2 shows statements in a modified JAVA source code file called Employee.java.")*;
- an XML type that provides XML-oriented data manipulation and implements a common Java™ type, wherein the XML type allows the combination of XML and Java™ type systems and is capable of accessing XML data from within Java without mapping the XML data to a Java™ object *(see Figure 1A: 116; Column 7: 31-38, "... the developer generates a modified JAVA source code file 112 based on the JAVA source code file 104 by inserting into a comment statement 114, for each class of data objects to be exchanged with other applications, data exchange configuration information in the form of XML configuration information 116 that indicates a relationship between data object classes and XML elements."; and 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5."; Column 8: 33-37, "Thus the XML configuration information 116 in comment 114 in the example of lines 3 through 4 indicates that the public class Employee is to be mapped to the root element "Employee" in the XML grammar defined in employee.dtd.")*; and

- an XML transformation capable of transformation a source type to a target type, wherein the source and target types can be either the XML type or the Java™ type system *(see Column 9: 26-31, "One or more methods of the user-defined doclet 124 produces statements 130 for an XML DTD or Schema document based on the XML configuration data 116 in the comment statement 114 for a data object in the modified source code file 112 and on the neighboring JAVA class definition statement.").*

As per **Claim 16**, the rejection of **Claim 15** is incorporated; and Hershberg further discloses:

- a global registry of XML transformations capable of looking up an existing XML transformation between a source and a target type *(see Column 11: 16-19, "In some of these embodiments, the conventional approach used employs not only the DTD file but also a mapping file that maps DTD elements/attributes to JAVA class attributes.").*

As per **Claim 17**, the rejection of **Claim 15** is incorporated; and Hershberg further discloses:

- a library of XML transformations capable of looking up an existing XML transformation by name between a source and a target type *(see Column 11: 16-19, "In some of these embodiments, the conventional approach used employs not only the DTD file but also a mapping file that maps DTD elements/attributes to JAVA class attributes.").*

As per **Claim 18**, Hershberg discloses:

- an XML data *(see Figure 1A: 118; Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118 ...")*;

- an XML schema which defines the XML data *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.")*;

- a lightweight XML store capable of retaining the XML data as a searchable index *(see Column 11: 16-19, "In some of these embodiments, the conventional approach used employs not only the DTD file but also a mapping file that maps DTD elements/attributes to JAVA class attributes.")*; and

- an XML type, which corresponds to the XML schema, that provides XML-oriented data manipulation and implements a common Java™ type, wherein the XML type allows the combination of XML and Java™ type systems and is capable of referencing the lightweight XML store and accessing elements of the XML data from within Java™ *(see Figure 1A: 116; Column 7: 31-38, "... the developer generates a modified JAVA source code file 112 based on the JAVA source code file 104 by inserting into a comment statement 114, for each class of data objects to be exchanged with other applications, data exchange configuration information in the form of XML configuration information 116 that indicates a relationship between data object classes and XML elements."; and 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in* *line 3.5."; Column 8: 33-37, "Thus the XML configuration information 116 in comment 114 in the example of lines 3 through 4 indicates that the public class Employee is to be mapped to the root element "Employee" in the XML grammar defined in employee.dtd.").*

As per **Claim 19**, Hershberg discloses:

- an XML data *(see Figure 1A: 118; Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118 ...")*;
- an XML schema which defines the XML data *(see Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.")*;
- a lightweight XML store capable of retaining the XML data at the text or tag level *(see Column 11: 16-19, "In some of these embodiments, the conventional approach used employs not only the DTD file but also a mapping file that maps DTD elements/attributes to JAVA class attributes.")*; and
- an XML type, which corresponds to the XML schema, that provides XML-oriented data manipulation and implements a common Java™ type, wherein the XML type allows the combination of XML and Java™ type systems and is capable of referencing the lightweight XML store and accessing elements of the XML data from within Java™ *(see Figure 1A: 116; Column 7: 31-38, "... the developer generates a modified JAVA source code file 112 based on the JAVA source code file 104 by inserting into a comment statement 114, for each class of data*

objects to be exchanged with other applications, data exchange configuration information in the form of XML configuration information 116 that indicates a relationship between data object classes and XML elements.”; and 54-58, “The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.”; Column 8: 33-37, “Thus the XML configuration information 116 in comment 114 in the example of lines 3 through 4 indicates that the public class Employee is to be mapped to the root element "Employee" in the XML grammar defined in employee.dtd.”).

As per **Claim 20**, the rejection of **Claim 19** is incorporated; and Hershberg further discloses:

- the lightweight XML store is capable of representing the retained XML data as a hierarchical structure *(see Column 13: 39-61).*

As per **Claim 21**, the rejection of **Claim 20** is incorporated; and Hershberg further discloses:

- the hierarchical structure can be a tree *(see Column 13: 39-61).*

As per **Claim 22**, the rejection of **Claim 19** is incorporated; and Hershberg further discloses:

- the XML type is capable of accessing the XML data incrementally *(see Column 7: 54-58, “The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-*

root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.").

Claims 23-36 are method claims corresponding to the computer-implemented system claims above (Claims 1-14) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-14.

Claims 37-39 are method claims corresponding to the computer-implemented system claims above (Claims 15-17) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 15-17.

Claim 40 is a method claim corresponding to the computer-implemented system claim above (Claim 18) and, therefore, is rejected for the same reason set forth in the rejection of Claim 18.

Claims 41-43 are method claims corresponding to the computer-implemented system claims above (Claims 19, 21, and 22) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 19, 21, and 22.

Claims 44-57 are machine readable medium claims corresponding to the computer-implemented system claims above (Claims 1-14) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-14.

Claims 58-60 are machine readable medium claims corresponding to the computer-implemented system claims above (Claims 15-17) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 15-17.

Claim 61 is a machine readable medium claim corresponding to the computer-implemented system claim above (Claim 18) and, therefore, is rejected for the same reason set forth in the rejection of Claim 18.

Claims 62-64 are machine readable medium claims corresponding to the computer-implemented system claims above (Claims 19, 21, and 22) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 19, 21, and 22.

Claim 65 is a system claim corresponding to the computer-implemented system claim above (Claim 1) and, therefore, is rejected for the same reason set forth in the rejection of Claim 1.

Response to Arguments

23. Applicant's arguments filed on May 6, 2008 have been fully considered, but they are not persuasive.

In the Remarks, Applicant argues:

a) Applicant respectfully submits that the user-defined doclet that is included in the JavaDoc process, as disclosed in Hershberg (Figures 1-2 and Column 9, Lines 21-32 and 56-57, and Table 3), does not implement a common Java type that provides XML-oriented data manipulation. As disclosed in Paragraph [0027] and [0032], a common Java type such as a (sic) XMLObject allows accessing and manipulating the XML data from within Java using the combination of XML and Java type systems.

Examiner's response:

a) Examiner disagrees. Applicant's arguments are not persuasive for at least the following reasons:

First, Hershberg clearly discloses "an XML type, which corresponds to the XML schema, that provides XML-oriented data manipulation and implements a common Java™ type" *(see Figure 1A: 116; Column 7: 54-58, "The XML configuration information 116 includes the user-defined JavaDoc tag "@xml-root-element" that is an example of tag 118, and the property "dtdFile." The property is set to a value "employee.dtd" given between the quotation marks in line 3.5.")*. Note that the XML configuration information (XML type that provides XML-oriented data manipulation) indicates that the class "Employee" (Java™ type) is to be mapped to the XML root element "Employee" defined in employee.dtd (XML schema).

Second, the claims recite only "XML type" with no further clarification on the claim scope of the term "type" as intended by the Applicant to cover. Thus, as the claims are interpreted as broadly as their terms reasonably allow (see MPEP § 2111.01 I), the interpretation of a broad limitation of "XML type" as XML configuration information and the like by one of ordinary skill in the art is considered to be reasonable by its plain meaning.

Third, Applicant is attempting to import limitations from the specification by referring to the specification. However, according to MPEP § 2111.01 II, it is improper to import claim limitations from the specification that are not part of the claims.

Fourth, contrary to the Applicant's assertion of a common Java™ type implemented as an "XMLObject," it is noted that paragraphs [0027] and [0032] of the specification describe that an XML type can be implemented as an "XMLObject," <u>NOT</u> a common Java™ type.

Conclusion

24. The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure.

25. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/QC/
July 29, 2008

/Wei Zhen/

Supervisory Patent Examiner, Art Unit 2191